**Esercizi da Risolvere con SQL:**

1. **Analisi delle Vendite Mensili:**
   * *Domanda:* Trova il totale delle vendite per ogni mese.

**Considerando importo transazione:**

SELECT

YEAR(DataTransazione) AS Anno,

MONTH(DataTransazione) AS Mese,

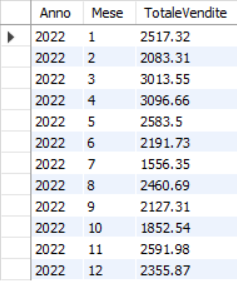
ROUND(SUM(ImportoTransazione), 2) AS TotaleVendite

FROM transazioni

GROUP BY YEAR(DataTransazione), MONTH(DataTransazione)

ORDER BY YEAR(DataTransazione), MONTH(DataTransazione);

RISULTATO:



**Considerando prezzo per quantità:**

SELECT

YEAR(DataTransazione) AS Anno,

MONTH(DataTransazione) AS Mese,

ROUND(SUM(p.Prezzo \* t.QuantitaAcquistata), 2) AS TotaleVendite

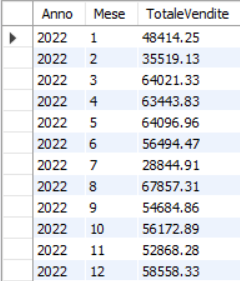
FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY YEAR(DataTransazione), MONTH(DataTransazione)

ORDER BY Anno, Mese;

RISULTATO:



1. **Prodotti più Venduti:**
   * *Domanda:* Identifica i tre prodotti più venduti e la loro quantità venduta.

SELECT

p.ProdottoID,

p.NomeProdotto AS Prodotto,

SUM(t.QuantitaAcquistata) AS QuantitaVenduta

FROM transazioni t

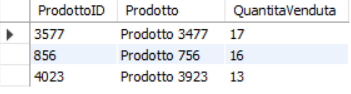
JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY p.ProdottoID, p.NomeProdotto

ORDER BY QuantitaVenduta DESC

LIMIT 3;

RISULTATO:



1. **Analisi Cliente:**
   * *Domanda:* Trova il cliente che ha effettuato il maggior numero di acquisti.

SELECT

ClienteID,

SUM(QuantitaAcquistata) AS TotaleAcquisti

FROM transazioni

GROUP BY ClienteID

ORDER BY TotaleAcquisti DESC

LIMIT 1;

RISULTATO:



1. **Valore medio della transazione:**
   * *Domanda:* Calcola il valore medio di ogni transazione.
   * *PER MESE:*

SELECT

YEAR(DataTransazione) AS Anno,

MONTH(DataTransazione) AS Mese,

ROUND(AVG(ImportoTransazione), 2) AS ValoreMedioTransazione

FROM transazioni

GROUP BY YEAR(DataTransazione), MONTH(DataTransazione)

ORDER BY Anno, Mese;

RISULTATO:



PER TRIMESTRE:

SELECT

YEAR(DataTransazione) AS Anno,

QUARTER(DataTransazione) AS Trimestre,

ROUND(AVG(ImportoTransazione), 2) AS ValoreMedioTransazione

FROM transazioni

GROUP BY YEAR(DataTransazione), Trimestre

ORDER BY Anno, Trimestre;

RISULTATO:



PER CATEGORIA DI PRODOTTO:

SELECT

p.Categoria,

ROUND(AVG(t.ImportoTransazione), 2) AS ValoreMedioTransazione

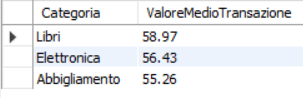
FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY p.Categoria

ORDER BY ValoreMedioTransazione DESC;

RISULTATO:



1. **Analisi Categoria Prodotto:**
   * *Domanda:* Determina la categoria di prodotto con il maggior numero di vendite.

SELECT

p.Categoria,

SUM(t.QuantitaAcquistata) AS QuantitaTotaleVenduta

FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY p.Categoria

ORDER BY QuantitaTotaleVenduta DESC

LIMIT 1;

RISULTATO:



1. **Cliente Fedele:**
   * *Domanda:* Identifica il cliente con il maggior valore totale di acquisti.

SELECT

t.ClienteID,

SUM(p.Prezzo \* t.QuantitaAcquistata) AS ValoreTotaleAcquisti

FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY t.ClienteID

ORDER BY ValoreTotaleAcquisti DESC

LIMIT 1;

RISULTATO:



1. **Spedizioni Riuscite:**
   * *Domanda:* Calcola la percentuale di spedizioni con "Consegna Riuscita".

TABELLA TRANSAZIONI:

SELECT

COUNT(\*) AS TotaleSpedizioni,

ROUND(SUM(CASE WHEN StatusConsegna = 'Consegna Riuscita' THEN 1 ELSE 0 END), 2) AS ConsegneRiuscite,

(SUM(CASE WHEN StatusConsegna = 'Consegna Riuscita' THEN 1 ELSE 0 END) / COUNT(\*)) \* 100 AS PercentualeConsegneRiuscite

FROM transazioni

WHERE StatusConsegna IS NOT NULL;

RISULTATO:



PRENDENDO TABELLA SPEDIZIONI:

SELECT

COUNT(\*) AS TotaleSpedizioni,

ROUND(SUM(CASE WHEN StatusConsegna = 'Consegna Riuscita' THEN 1 ELSE 0 END), 2) AS ConsegneRiuscite,

(SUM(CASE WHEN StatusConsegna = 'Consegna Riuscita' THEN 1 ELSE 0 END) / COUNT(\*)) \* 100 AS PercentualeConsegneRiuscite

FROM spedizioni

WHERE StatusConsegna IS NOT NULL;

RISULTATO:



1. **Prodotto con la Migliore Recensione:**
   * *Domanda:* Trova il prodotto con la recensione media più alta.

PER CATEGORIA PRODOTTO:

SELECT

p.Categoria,

AVG(r.Rating) AS MediaRecensioni

FROM ratings\_dataset r

JOIN prodotti p ON r.ProductID = p.ProdottoID

GROUP BY p.Categoria;

RISULTATO:



(CERCARE DI MIGLIORARE LA QUALITA DEI PRODOTTI)

1. **Analisi Temporale:**
   * *Domanda:* Calcola la variazione percentuale nelle vendite rispetto al mese precedente.

SELECT

Anno,

Mese,

TotaleVendite,

LAG(TotaleVendite) OVER (ORDER BY Anno, Mese) AS VenditeMesePrecedente,

((TotaleVendite - LAG(TotaleVendite) OVER (ORDER BY Anno, Mese)) / LAG(TotaleVendite) OVER (ORDER BY Anno, Mese)) \* 100 AS VariazionePercentuale

FROM (

SELECT

YEAR(DataTransazione) AS Anno,

MONTH(DataTransazione) AS Mese,

SUM(p.Prezzo \* t.QuantitaAcquistata) AS TotaleVendite

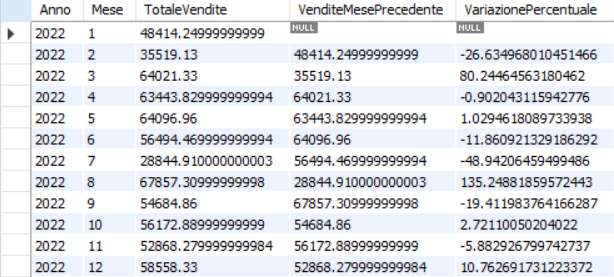
FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY Anno, Mese

) AS VenditeMensili;

RISULTATO:



1. **Quantità di Prodotti Disponibili:**
   * *Domanda:* Determina la quantità media disponibile per categoria di prodotto.

SELECT

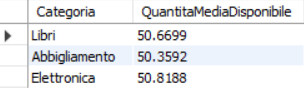
Categoria,

AVG(QuantitaDisponibile) AS QuantitaMediaDisponibile

FROM prodotti

GROUP BY Categoria;

RISULTATO:



1. **Analisi Spedizioni:**
   * *Domanda:* Trova il metodo di spedizione più utilizzato.

SELECT

MetodoSpedizione,

COUNT(\*) AS QuantitaSpedizioni

FROM spedizioni

GROUP BY MetodoSpedizione

ORDER BY QuantitaSpedizioni DESC

LIMIT 1;

RISULTATO:



1. **Analisi dei Clienti:** 
   * *Domanda:* Calcola il numero medio di clienti registrati al mese.

SELECT

Anno,

Mese,

AVG(NumeroClientiRegistrati) AS NumeroMedioClientiRegistrati

FROM (

SELECT

YEAR(DataRegistrazione) AS Anno,

MONTH(DataRegistrazione) AS Mese,

COUNT(\*) AS NumeroClientiRegistrati

FROM clienti

GROUP BY Anno, Mese

) AS ClientiPerMese

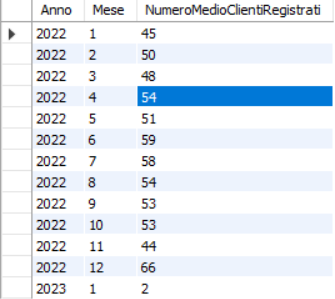
GROUP BY Anno, Mese

ORDER BY Anno, Mese;

RISULTATO:



RISULTATO SENZA .0000:



1. **Prodotti Rari:**
   * *Domanda:* Identifica i prodotti con una quantità disponibile inferiore alla media.

Per Categoria:

SELECT

Categoria,

AVG(QuantitaDisponibile) AS QuantitaMediaCategoria

FROM prodotti

GROUP BY Categoria

HAVING AVG(QuantitaDisponibile) < (

SELECT AVG(QuantitaDisponibile) AS MediaGenerale FROM prodotti

);

RISULTATO:



SCRIPT CON + E -:

SELECT

NomeProdotto,

Categoria,

QuantitaDisponibile,

CASE

WHEN QuantitaDisponibile > (SELECT AVG(QuantitaDisponibile) FROM prodotti) THEN '+'

WHEN QuantitaDisponibile < (SELECT AVG(QuantitaDisponibile) FROM prodotti) THEN '-'

ELSE '='

END AS ConfrontoConMedia

FROM

Prodotti

ORDER BY Categoria;

SCRIPT CON CONFRONTO CATEGORIA:

SELECT Categoria, AVG(QuantitaDisponibile) AS MediaQuantitaCategoriaFROM prodottiGROUP BY Categoria;SELECT NomeProdotto, Categoria, QuantitaDisponibile, AVG(QuantitaDisponibile) OVER (PARTITION BY Categoria) AS MediaQuantitaCategoria, CASE WHEN QuantitaDisponibile > AVG(QuantitaDisponibile) OVER (PARTITION BY Categoria) THEN '+' WHEN QuantitaDisponibile < AVG(QuantitaDisponibile) OVER (PARTITION BY Categoria) THEN '-' ELSE '=' END AS ConfrontoConMediaFROM prodottiORDER BY Categoria, NomeProdotto;

SCRIPT:

SELECT

ProdottoID,

NomeProdotto,

QuantitaDisponibile

FROM

prodotti

WHERE

QuantitaDisponibile < (SELECT AVG(QuantitaDisponibile) FROM prodotti);

1. **Analisi dei Prodotti per Cliente:**
   * *Domanda:* Per ogni cliente, elenca i prodotti acquistati e il totale speso.

SELECT

a.ClienteID,

a.ProdottoID,

ROUND(SUM(a.QuantitaAcquistata \* b.Prezzo), 2) AS Spesa\_acquisti

FROM transazioni a

JOIN prodotti b ON a.ProdottoID = b.ProdottoID

GROUP BY a.ClienteID, a.ProdottoID

ORDER BY a.ClienteID ASC;

SCRIPT CON QUANTITà ACQUISTATA:

SELECT

a.ClienteID,

a.ProdottoID,

ROUND(SUM(a.QuantitaAcquistata \* b.Prezzo), 2) AS Spesa\_acquisti,

MAX(a.QuantitaAcquistata) AS MaxQuantitaAcquistata

FROM transazioni a

JOIN prodotti b ON a.ProdottoID = b.ProdottoID

GROUP BY a.ClienteID, a.ProdottoID

ORDER BY a.ClienteID ASC;

1. **Miglior Mese per le Vendite:**
   * *Domanda:* Identifica il mese con il maggior importo totale delle vendite.

SELECT

YEAR(DataTransazione) AS Anno,

MONTH(DataTransazione) AS Mese,

ROUND(SUM(p.Prezzo \* t.QuantitaAcquistata), 2) AS ImportoTotaleVendite

FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY Anno, Mese

ORDER BY ImportoTotaleVendite DESC

LIMIT 1;

RISULTATI:



1. **Analisi dei Prodotti in Magazzino:**
   * *Domanda:* Trova la quantità totale di prodotti disponibili in magazzino.

SELECT SUM(QuantitaDisponibile) AS QuantitaTotaleDisponibile

FROM prodotti;

RISULTATO:



1. **Clienti Senza Acquisti:**
   * *Domanda:* Identifica i clienti che non hanno effettuato alcun acquisto.

PER AVERE I CLIENTI:

SELECT c.ClienteID, c.NomeCliente, c.Email

FROM clienti c

LEFT JOIN transazioni t ON c.ClienteID = t.ClienteID

WHERE t.ClienteID IS NULL;

CONTEGGIO CLIENTI:

SELECT COUNT(\*) AS NumeroClientiSenzaAcquisti

FROM clienti c

LEFT JOIN transazioni t ON c.ClienteID = t.ClienteID

WHERE t.ClienteID IS NULL;

RISULTATO:



CONTEGGIO CLIENTI CON TOTALI:

SELECT

MAX(NumeroClientiTransazioni) AS NumeroClientiTransazioni,

MAX(NumeroClientiClienti) AS NumeroClientiClienti,

MAX(ClientiSoloInTransazioni) AS ClientiSoloInTransazioni,

MAX(ClientiSoloInClienti) AS ClientiSoloInClienti

FROM (

SELECT

COUNT(DISTINCT t.ClienteID) AS NumeroClientiTransazioni,

0 AS NumeroClientiClienti,

COUNT(DISTINCT CASE WHEN c.ClienteID IS NULL THEN t.ClienteID END) AS ClientiSoloInTransazioni,

0 AS ClientiSoloInClienti

FROM

transazioni t

LEFT JOIN

clienti c ON t.ClienteID = c.ClienteID

UNION ALL

SELECT

0 AS NumeroClientiTransazioni,

COUNT(DISTINCT c.ClienteID) AS NumeroClientiClienti,

0 AS ClientiSoloInTransazioni,

COUNT(DISTINCT CASE WHEN t.ClienteID IS NULL THEN c.ClienteID END) AS ClientiSoloInClienti

FROM

transazioni t

RIGHT JOIN

clienti c ON t.ClienteID = c.ClienteID

) AS Result;

RISULTATO:



1. **Analisi Annuale delle Vendite:**
   * *Domanda:* Calcola il totale delle vendite per ogni anno.

SELECT

YEAR(DataTransazione) AS Anno,

ROUND(SUM(p.Prezzo \* t.QuantitaAcquistata), 2) AS TotaleVendite

FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

GROUP BY Anno

ORDER BY Anno;

RISULTATO:



1. **Spedizioni in Ritardo:**
   * *Domanda:* Trova la percentuale di spedizioni con "In Consegna" rispetto al totale.

SELECT (COUNT(StatusConsegna)/5000)\*100 AS percentuale\_ritardoFROM spedizioniWHERE StatusConsegna = 'In Consegna';

RISULTATO:



1. QUERY PER TROVARE LA MEDIA RATING PER MESE E CATEGORIA.

SELECT

YEAR(t.DataTransazione) AS Anno,

MONTH(t.DataTransazione) AS Mese,

p.Categoria,

SUM(p.Prezzo \* t.QuantitaAcquistata) AS EntrateGenera,

AVG(r.Rating) AS MediaRating

FROM transazioni t

JOIN prodotti p ON t.ProdottoID = p.ProdottoID

LEFT JOIN ratings\_dataset r ON t.ProdottoID = r.ProductID

GROUP BY Anno, Mese, p.Categoria;

RISULTATI:



